IN THE UNITED STATES
PATENT AND TRADEMARK OFFICE

Patent Application

Inventor	**J.A. Ford**
Case	**Ford 1**
Serial No.	**09/420,912** Group Art Unit: **3623**
Filing Date	**October 20, 1999**
Examiner	**R. Bachner**
Title	**Arrangement for Resource and Work-Item Selection**

DIRECTOR FOR PATENTS
WASHINGTON, DC 20231

SIR:

DELCARATION UNDER 37 C.F.R. §1.131

I, David Volejnicek, declare as follows:

1. The above-identified inventor ("Inventor") is no longer employed by either Lucent Technologies, Inc., the original assignee of the above-identified application ("Application"), or Avaya Inc., the successor-in-interest to the Application, and his whereabouts are unknown to me.

2. I prepared and filed the Application in my capacity as the Inventor's patent attorney.

3. The Inventor conceived of the invention at least by October 30, 1998, as evidenced by Attachment A hereto, which is the Inventor's memorandum describing the invention.

4. The disclosure of Attachment A substantially corresponds to the disclosure of page 1, lines 11-31; page 6, line 3, to page 13, line 4; and page 13,

line 20, to page 16, line 20, of the Application, as evidenced by comparison of those two documents.

5. The Inventor met with me to discuss patenting of the subject matter disclosed in Attachment A on November 16, 1998, and I recommended that a request to approve patenting be submitted by him to the Lucent Patent Committee, both as evidenced by Attachment B hereto, which are my notes of the meeting.

6. The Inventor requested approval of patenting on January 29, 1999, by means of a Patent Submission to the Lucent Patent Committee which forms Attachment C hereto.

7. The Lucent Patent Committee evaluated the request at its meeting of February 11,1999, as evidenced by Attachment D hereto which is a binder divider showing which submissions were considered at the February 11, 1999 meeting, and approved the idea for patenting, as evidenced by the existence of the Application.

8. The Inventor continued to work on and improve the invention, as evidenced by Attachment E hereto which is an April 15, 1999, expanded revision of Attachment A.

9. A prototype implementation was created and tested by the Inventor by March of 1999, as evidenced by Attachment F hereto which is a summary of activities pertaining to the invention that was prepared by the Inventor at the time of filing of the Application.

10. Subsequently to February 11, 1999, until the filing date of the Application, I worked with the inventor to prepare and file the Application. I commenced work on the Application no later than June 24, 1999, as evidenced by Attachments G and H hereto, which are documents prepared during the preparation of the application.

11. My normal procedure in preparing and filing a patent application involves meeting with the inventor one or more times to discuss the invention and to obtain any information supplementing the previously-written documents such as Attachments A, C, and E, then drafting a rough draft of the application, submitting the draft to the inventor for review and comment, redrafting the application in light of the inventor's comments and resubmitting the redraft to the inventor, then repeating the previous step if and as often as necessary, and, upon the inventor's approval of the application, filing the application in the U.S. Patent and Trademark Office. Although meeting notes and drafts of the application are no longer available, I believe that I also followed the above-described process in preparing and filing this Application.

12. The Application was filed with the U.S. Patent and Trademark Office on October 20, 1999, which is within one year of the December 8, 1998, filing date of U.S. Pat. No. US 6,275,812 B1 (Haq et al.).

13. I acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both (19 U.S.C. §1001) and may jeopardize the validity of the application or any patent issuing thereon. All statements made of my own knowledge are true and all statements made on information and belief are believed to be true.

By David Volejnicek

David Volejnicek
Reg. No. 29355

Date: 28 Jan. 20 03

Avaya Inc.
Docket Administrator
307 Middletown-Lincroft Road
Room 1N-391
Lincroft, NJ 07738

Enclosures: Attachments A-H

A New Algorithm for Resource and Contact Selection

Jon Ford

October 30, 1998

1 Overview

Moving resource and contact selection algorithms off the switch has opened the door to many new opportunities for higher optimization and flexibility.

2 Current Limitations

2.1 Hardware Limitations

Traditional call center resource and contact selection algorithms have been limited by the *call center model* available in the switch. For example, Definity uses resources with skills and skill levels. The model then provides multi-priority queues for queueing contacts to wait for the resources with those skills to become available. So, any alogorithms we apply to this model are limited by its unflexible and simple structure. The limitations become more obvious when we look at an example in the following section.

2.2 Example - Acme Insurance

Supose that Acme Insurance Company sells 3 types of insurance: auto, home, and life. Also, suppose that each resource must be licenced to sell each of these insurance policies on a state by state basis. Lastly, suppose that the call center allows callers to select English or Spanish speaking resources. This results in $3 \times 50 \times 2 = 300$ possible combinations of caller skill requirements. Administering resources and skills using these combinations would be difficult at best. In fact, given the current limitations with market leading ACDs, a work-around must be found.

3 A New Approach

Separated from the constraints of the legacy switching systems, we are free to apply *common sense* to our way of thinking about resources, skills, and callers.

We will define resources with skills and proficiencies in those skills. We will qualify calls (contacts) when they arrive in the call center as requiring sets of skills. We will set goals for contact handling and resource treatment. Lastly, we will optimize and allow extensive customization of the process of matching the resources with the contacts - even on a contact or resource specific basis.

4 The Algorithm

The goal of the algorithm is to match contacts and resources in such a way that it brings the most value to all of the stakeholders in the call center organization. This includes customers, resources, managers, etc. To perform this optimization, we will evaluate three components of the contact-resource match.

4.1 The Three Call Center Values

4.1.1 Business Value

The *Business Value* of a contact-resource match is a measure of resource qualification for contact handling based on resource skills and skill proficiencies and contact skill requirements.

4.1.2 Resource Treatment Value

The *Resource Treatment Value* of a contact-resource match is a measure of how an resource is spending time compared with other resources as well as individual resource goals.

4.1.3 Contact Treatment Value

The *Contact Treatment Value* of a contact-resource match is a measure of how a contact is treated compared to other contacts as well as individual treatement goals for the specific contact.

4.2 The Two Basic Call Center States

Evaluation of the Three Call Center Values discussed above depends on the state of the call center at the time of evalution. The two possible states for a call center and there effect on the applicability of the Three Call Center Values is described in the following sections.

4.2.1 Resource Surplus

If there are resources working in the call center who are able to handle contacts but are waiting for contact, the call center is considered to be in a state of *Resource Surplus*. In the case of resource surplus, when a contact arrives in the call center, the objective is to find the *best* resource to handle the contact. So, we would evaluate the Business Value and the Resource Treatment Value of each

resource matched with the new contact, looking for the resource-contact match that brought the highest value to the organization. For this case of Resource Surplus, Contact Treatment is not an issue as the contact will receive immediate handling, provided that there is an resource available with the required skills.

4.2.2 Contact Surplus

If there are no resources in the call center who are able to handle contacts and there are contacts waiting for resource assignment, the call center is considered to be in a state of *Contact Surplus*. In the case of contact surplus, when an resource becomes available to handle contacts, the objective is to find the *best* contact for the resource to handle. So, we would evaluate the Business Value and the Contact Treatment value of each contact matched with the newly available resource, looking for the resource-contact match that brings the hightest value to the organization. For the case of Resource Surplus, Resource Treatment is not an issue as the resource will receive the contact immediately upon becoming available, provided that there is a contact available for which the resource has the required skills.

4.3 Calculation of Business Value

Assume that matrix $A_{availresources,maxskills}$, where $availresources$ is the number of available resources and $maxskills$ is the maximum number of skills defined in the call center, represents the available resources and their associated skill levels such that $A_{n,m}$ represents the skill level (an integer between 0 and 10) for resource n and skill m.

Also, assume that matrix $BR_{pendingcontacts,maxskills}$, where $pendingcontacts$ is the number of unserviced contacts in the call center and $maxskills$ is the maximum number of skills defined in the call center, represents the skill *weight* of the contact such that $BR_{n,m}$ represents the skill *weight* (an integer between 0 and 10) for contact n and skill m.

Lastly, assume that matrix $BRR_{pendingcontacts,maxskills}$, where $pendingcontacts$ is the number of unserviced contacts in the call center and $maxskills$ is the maximum number of skills defined in the call center, represents the *requirement* of a skill for a contact such that $BRR_{n,m}$ (true of false) indicates whether an resource *must* have a skill level > 0 in skill m to handle contact n. This array is not used in the calculation of business value, but it will be used in following sections.

4.3.1 Resource Surplus

In the case of Resource Surplus Call Center State, the business value is calculated as:

$$BusinessValue_{Resource_n} = \sum_{i=1}^{maxskills} (A_{n,i} \times BR_{1,i}) = ASBV_n$$

4.3.2 Contact Surplus

In the case of Contact Surplus Call Center State, the business value is calculates as:

$$BusinessValue_{Contact_n} = \sum_{i=1}^{maxskills} (A_{1,n} \times BR_{n,i}) = CSBV_n$$

4.4 Calculation of Resource Treatment Value

Assume that the matrix $T_{availresources,3}$, where $availresources$ is the number of resources in the call center waiting to service contacts, represents the resource's activity during the current login session by $T_{n,1}$ = time in seconds since becoming available, $T_{n,2} = 100$ - percent of logged in time spent handling contacts and their contacts associated after call work, and $T_{n,3}$ = a measure of how much handling the current contact would move them toward their goal (See section on calculation and setting of resource goals.)

Next, assume that matrix $TW_{pendingcontacts,3}$ represents the weight to be applied to each of the treatment values such that $TW_{n,m}$ is the weight for resource n to be applied to $T_{n,m} \forall m, where 1 \leq m \leq 3$.

Now, we calculate the Resource Treatment value as:

$$ResourceTreatmentValue_{Resource_n} = \sum_{i=1}^{3} (T_{n,i} \times TW_{n,i}) = ATV_n$$

4.5 Calculation of Contact Treatment Value

Assume that the matrix $C_{pendingcontacts,3}$, where $pendingcontacts$ is the number of contacts awaiting service in the call center, represents the contact's current waiting condition by $C_{n,1}$ = number of seconds the contact has been waiting, $C_{n,2}$ = the estimated time that the contact will have to wait for service, and $C_{n,3}$ = the number of seconds the contact is over its acceptable wait time (or zero if not over acceptable wait time.)

Also, assume that matrix $CW_{pendingcontacts,3}$ represents the weight to be applied to each of the treatment values such that TWn, m is the weight for resource n to be applied to $T_{n,m} \forall m, where 1 \leq m \leq 3$.

Now we calculate the Contact Treatment value as:

$$ContactTreatmentValue_{Contact_n} = \sum_{i=1}^{3} (C_{n,i} \times CW_{n,i}) = CTV_n$$

5 Making the Decision

One final thing remains for resource or contact selection. The wieght of the business value vs. Resource Treatment value, in the case of resource surplus, and the weight of business value vs. Contact Treatment in case of contact surplus.

We will define W_{ASBV} = weight of business value with resource surplus, W_{AT} = weight of Resource Treatment value with resource surplus, W_{CSBV} = weight of business value with contact surplus, and W_{CV} = weight of contact treatment value in the case of contact surplus.

5.1 Resource Surplus and Resource Selection

For the case of resource surplus, we seek to maximize the business value and resource treatment value according to the specified weights by selecting the resource who scores the highest as follows:

$$Max\left[(ASBV_n \times W_{ASBV}) + (ATV_n \times W_{AT})\right], where A_{n,i} > 0\ \forall BRR_{1,i} > 0.$$

5.2 Contact Surplus and Contact Selection

For the case of contact surplus, we seek to maximize the business value and contact treatment value according to the specified weights by selecting the contact which scores the highest as follows:

$$Max\left[(CSBV_n \times W_{CSBV}) + (CTV_n \times W_{CV})\right], where A_{1,i} > 0\ \forall BRR_{n,i} > 0.$$

6 Scaling Results for Practical Application

The above algorithms select an resource or a contact based on a highest score. However, determining the weights of each of the values that result in an optimum solution for the call center can be impractical. For instance, if an resource scores very high in business value ($10 \times 10 = 100$) but the contacts have been waiting for an average of 5 minutes, the contact treatment value (CTV) could be as high as 300. So, as the contacts age, the CTV has an increased effect on the overall selection. This is undesireable.

So, to be ***practically applicable*** we need to scale the calculation of each of the three selection values. We will need a scaling value for each of the calculations that are performed either for all resources or all contacts as follows (denoting scaling values as S):

$$S_{ASBV} = \frac{100}{Max[ASBV_n]}$$

$$S_{CSBV} = \frac{100}{Max[ASBV_n]}$$

$$S_{T_i} = \frac{100}{Max[T_{n,i}]}, 1 \leq i \leq 3$$

$$S_{C_i} = \frac{100}{Max[C_{n,i}]}, 1 \leq i \leq 3$$

$$S_T = \frac{100}{Max[T_n]}$$

$$S_C = \frac{100}{Max[C_n]}$$

Now, we can express the weighted values for $ASBV$, $CSBV$, ATV, and CTV as follows:

$$WeightedBusinessValue_{Resource_n} = S_{ASBV} ASBV_n = WASBV_n$$

$$BusinessValue_{Contact_n} = S_{CSBV} CSBVn = WCSBV_n$$

$$PartialWeightedResourceTreatmentValue_{Resource_n} = \sum_{i=1}^{3} (S_{T_i} \times T_{n,i} \times TW_{n,i}) = PWATV_n$$

$$PartialWeightedContactTreatmentValue_{Contact_n} = \sum_{i=1}^{3} (S_{C_i} \times C_{n,i} \times CW_{n,i}) = PWCTV_n$$

$$WeightedResourceTreatmentValue_{Resource_n} = PWATV_n S_T = WATV_n$$

$$WeightedContactTreatmentValue_{Contact_n} = PWCTV_n S_C = WCTV_n$$

The net result is that we finish the calculation with each of our values ($WASBV$, $WCSBV$, $WATV$, and $WCTV$ between 0 and 100. Now, regardless of the units of measure or scaling internal to each of the value's initial configuration we can consistently express the weight that these values have on the overall resource and contact selection.

$$Max\left[(WASBV_n \times W_{ASBV}) + (WATV_n \times W_{AT})\right], where A_{n,i} > 0 \, \forall BRR_{1,i} > 0.$$

$$Max\left[(WCSBV_n \times W_{CSBV}) + (WCTV_n \times W_{CV})\right], where A_{1,i} > 0 \, \forall BRR_{n,i} > 0.$$

Note that the weighting for the components of Resource Treatment Value ($T_{n,i}$) and Contact Treatment Value ($C_{n,i}$) are scaled by the above calculations as well.

7 An Example

We will look at an example of resource surplus (Resource Selection) contact-resource matching to clarify the ideas expressed above. Assume that we have two resources, Jane and Jeff. And, we have skills 1-6, English, Chinese, Tech-support, Sales, PC, and UNIX. Jeff speaks English well (skill level 10), speaks Chinese (skill level 5), has been trainied in tech support (skill level 8) and UNIX(skill level 10). Jane speaks English(skill level 10), and is trained and experienced in Sales (skill level 10), she is just learning tech support (skill level 2) and has experience in PC (skill level 7) and UNIX (skill level 6). Also, assume that both Jeff and Jane have logged in and are waiting to receive contacts.

Now, our matrix A of available resources looks like this:

$$A = \begin{bmatrix} 10 & 5 & 8 & 0 & 0 & 10 \\ 10 & 0 & 2 & 10 & 7 & 6 \end{bmatrix}$$

Further assume that Jeff has been idle for 2 minutes and has been on calls 75% of the day. Jane has been idle for 3 minutes and on calls 50% of the day. Now our matrix T looks like this:

$$T = \begin{bmatrix} 60 & 25 & 0 \\ 180 & 50 & 0 \end{bmatrix}$$

Now, a contact arrives in the call center. The contact is qualified with the following requirements: English(weight 10), Tech support (weight 10), and UNIX (weight 5.) All three of these skills are *required*. Now, our matrix BR and matrix BRR look like this:

$$BR = \begin{bmatrix} 10 & 0 & 10 & 0 & 0 & 5 \end{bmatrix}$$

$$BRR = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Lastly, when the contact arrives, it is assigned the following weights:

$$TW = \begin{bmatrix} .25 & .75 & 0 \end{bmatrix}$$

$$W_{ASBV} = .2$$

$$W_{AT} = .8$$

Now, we can calculate our best selection...

$$ASBV_{Jeff} = 230$$

$$ASBV_{Jane} = 150$$

$$WASBV_{Jeff} = 230\frac{100}{230} = 100$$

$$WASBV_{Jane} = 150\frac{100}{230} = 65.22$$

$$PWATV_{Jeff} = \frac{100}{180}(60)(.25) + \frac{100}{50}(25)(.75) + 0 = 45.83$$

$$PWATV_{Jane} = \frac{100}{180}(180)(.25) + \frac{100}{50}(50)(.75) + 0 = 200$$

$$WATV_{Jeff} = \frac{100}{200}(45.83) = 22.92$$

$$WATV_{Jane} = \frac{100}{200}200 = 100$$

Total Value for Jeff:

$$100(.2) + 22.92(.8) = 40.33$$

Total Value for Jane:

$$65.22(.2) + 100(.8) = 93.044$$

Therefore, Jane is selected to handle the contact. Note that Jeff scored higher in business value and Jane scored higher in agent fairness value. However, due to the 80 % weight on the Resource Fairness value, the contact was delivered to Jane.

John Ford

(Dave Bauer)

11-16-98

Attachment B

"Soft ACD" functionality in off-board solution

contact = call

no commercial date

may be in Center View Indirect (Bullwinkle)

P.A.:

Bogart 17

Gabriel 1

Tonisson 1

I asked for submission

DR-PAT-99-18

Attachment C

RECEIVED
FEB 1 1999
LEGAL DEPARTMENT
DENVER

Lucent Technologies
Bell Labs Innovations

ject: **Patent Submission – Resource and Work Item Selection**

date: **January 29, 1999**

from: **Jon Ford**
139KE0000
DR 1C49
80881

TO: BCS-Denver Patent Committee – Room 2U41, 11900 North Pecos Street, Denver, CO. 80234-2703

Problem:

The idea solves the problem of distribution of work items to resources by selecting either the best resource for a work item, in the case of multiple available resources, or selecting the best work item, in the case of multiple available work items.

Prior Art:

Traditional methods employ first in first out (FIFO) queuing (the first available resource handles the first available work item.) Lucent has expanded on this idea with the Advocate algorithms and their application to the specific instance of work item and resource ching involving agents and incoming calls in call centers. There are also a number of other companies providing varying levels of resource – work item matching involving some form of queuing or FIFO treatment.

Brief Description:

See attached document "A New Algorithm for Resource and Work Item Selection."

Comparison:

This solution differs from prior art in the following ways:

1. Allows for an unlimited number of skills (limited by computing power only)
2. Individual weighting of multiple skills required to process each work item
3. Consideration of both required and non-required skills
4. Consideration and weighting of multiple skill levels across multiple skills.
5. Each work item can receive a different priority weight for business value, resource fairness, and work item fairness.
6. Each work item can receive a different priority weight for different categories of resource fairness.
7. Each work item can receive a different priority weight for different categories of work item fairness.
8. An estimated "time to service" is calculated for calls requiring multiple skills for service based on the weighting of those skill requirements, the number of work items waiting for service and their skill requirement weighting, and "time to service" of work items already distributed to resources.
9. The above "time to service" calculation is based on a single vector allowing for easy distribution of workflow across multiple sites and centralization of the site selection decision.

Use of the Idea:

This idea would be used in Lucent Technologies products initially in the distribution of calls, email, and other media contacts in call centers. As our products grow to encompass more general workflow concepts, the idea will be applicable in that area as well.

Detection of Use:

Administration of the work item distribution system would indicate very clearly the use of the idea.

Economic Impact:

Unknown.

Originators:

Jon Ford
139KE0000
1C-49
303-538-0881
jonford@lucent.com

Classification:
Your Rating ? ____II_____
Your Department Head's Name and Rating ? _Lucy Sanders - II_________

Attorney:

David Volejnicek

Attachments:

A New Algorithm for Resource and Work Item Selection. Jon Ford, Lucent Technologies.

A New Algorithm for Resource and Work Item Selection

Jon Ford, Lucent Technologies

January 29, 1999

1 Overview

Moving resource and contact selection algorithms off the switch has opened the door to many new opportunities for higher optimization and flexibility. The algorithm described below was initially developed for distributing calls in a call center. However, as the ideas were developed it became more and more obvious that these ideas apply to any kind of work distribution. So, the algorithm described below is a more general solution to what started out as a specific problem.

2 Current Limitations

2.1 Hardware Limitations

Traditional call center resource and work item selection algorithms have been limited by the *call center model* available in the switch. For example, Definity uses resources with skills and skill levels. The model then provides multi-priority queues for queueing work items to wait for the resources with those skills to become available. So, any algorithms we apply to this model are limited by its unflexible and simple structure. The limitations become more obvious when we look at an example in the following section.

2.2 Example - Acme Insurance

Supose that Acme Insurance Company sells 3 types of insurance: auto, home, and life. Also, suppose that each resource must be licenced to sell each of these insurance policies on a state by state basis. Lastly, suppose that the call center allows callers to select English or Spanish speaking resources. This results in $3 \times 50 \times 2 = 300$ possible combinations of caller skill requirements. Administering resources and skills using these combinations would be difficult at best. In fact, given the current limitations with market leading ACDs, a work-around must be found.

3 A New Approach

Separated from the constraints of the legacy switching systems, we are free to apply *common sense* to our way of thinking about resources, skills, and callers. We will define resources with skills and proficiencies in those skills. We will qualify calls (work items) when they arrive in the system as requiring sets of skills. We will set goals for work item handling and resource treatment. Lastly, we will optimize and allow extensive customization of the process of matching the resources with the work items - even on a work item or resource specific basis.

4 The Algorithm

The goal of the algorithm is to match work items and resources in such a way that it brings the most value to all of the stakeholders in the call center organization. This includes customers, resources, managers, etc. To perform this optimization, we will evaluate three components of the work item-resource match.

4.1 The Three System Values

4.1.1 Business Value

The *Business Value* of a work item-resource match is a measure of resource qualification for work item handling based on resource skills and skill proficiencies and work item skill requirements.

4.1.2 Resource Treatment Value

The *Resource Treatment Value* of a work item-resource match is a measure of how an resource is spending time compared with other resources as well as individual resource goals.

4.1.3 Work Item Treatment Value

The *Work Item Treatment Value* of a work item-resource match is a measure of how a work item is treated compared to other work items as well as individual treatement goals for the specific work item.

4.2 The Two Basic System States

Evaluation of the Three System Values discussed above depends on the state of the system at the time of evalution. The two possible states for a system and their effect on the application of the Three System Values is described in the following sections.

4.2.1 Resource Surplus

If there are resources working in the system who are able to handle work items but are waiting for a work item, the system is considered to be in a state of *Resource Surplus.* In the case of resource surplus, when a work item arrives in the system, the objective is to find the *best* resource to handle the work item. So, we would evaluate the Business Value and the Resource Treatment Value of each resource matched with the new work item, looking for the resource-work item match that brought the highest value to the organization. For this case of Resource Surplus, Work Item Treatment is not an issue as the work item will receive immediate handling, provided that there is an resource available with the required skills.

4.2.2 Work Item Surplus

If there are no resources in the system who are able to handle work items and there are work items waiting for resource assignment, the system is considered to be in a state of *Work Item Surplus.* In the case of work item surplus, when a resource becomes available to handle work items, the objective is to find the *best* work item for the resource to handle. So, we would evaluate the Business Value and the Work Item Treatment value of each work item matched with the newly available resource, looking for the resource-work item match that brings the hightest value to the organization. For the case of Resource Surplus, Resource Treatment is not an issue as the resource will receive the work item immediately upon becoming available, provided that there is a work item available for which the resource has the required skills.

4.3 Calculation of Business Value

Assume that matrix $A_{availresources,maxskills}$, where $availresources$ is the number of available resources and $maxskills$ is the maximum number of skills defined in the system, represents the available resources and their associated skill levels such that $A_{n,m}$ represents the skill level (an integer between 0 and 10) for resource n and skill m.

Also, assume that matrix $BR_{pendingworkitems,maxskills}$, where $pendingworkitems$ is the number of unserviced work items in the system and $maxskills$ is the maximum number of skills defined in the system, represents the skill *weight* of the work item such that $BR_{n,m}$ represents the skill *weight* (an integer between 0 and 10) for work item n and skill m.

Lastly, assume that matrix $BRR_{pendingworkitems,maxskills}$, where $pendingworkitems$ is the number of unserviced work items in the system and $maxskills$ is the maximum number of skills defined in the system, represents the *requirement* of a skill for a work item such that $BRR_{n,m}$ (true of false) indicates whether a resource *must* have a skill level > 0 in skill m to handle work item n. This array is not used in the calculation of business value, but it will be used in following sections.

4.3.1 Resource Surplus

In the case of Resource Surplus System State, the business value is calculated as:

$$BusinessValue_{Resource_n} = \sum_{i=1}^{maxskills} (A_{n,i} \times BR_{1,i}) = RSBV_n$$

4.3.2 Work Item Surplus

In the case of Work Item Surplus System State, the business value is calculates as:

$$BusinessValue_{WorkItem_n} = \sum_{i=1}^{maxskills} (A_{1,n} \times BR_{n,i}) = WSBV_n$$

4.4 Calculation of Resource Treatment Value

Assume that the matrix $T_{availresources,3}$, where $availresources$ is the number of resources in the system waiting to service work items, represents the resource's activity during the current login session by $T_{n,1}$ = time in seconds since becoming available, $T_{n,2}$ = 100 - percent of logged in time spent handling work items and their work items associated after call work, and $T_{n,3}$ = a measure of how much handling the current work item would move them toward their goal (See section on calculation and setting of resource goals.)

Next, assume that matrix $TW_{pendingworkitems,3}$ represents the weight to be applied to each of the treatment values such that $TW_{n,m}$ is the weight for resource n to be applied to $T_{n,m} \forall m, where 1 \leq m \leq 3$.

Now, we calculate the Resource Treatment value as:

$$ResourceTreatmentValue_{Resource_n} = \sum_{i=1}^{3}(T_{n,i} \times TW_{n,i}) = RTV_n$$

4.5 Calculating and Setting Resource Goals

Let $G_{maxskills,numresources}$ be a vector containing percent allocation goals for all resources up to maxskills such that

$$\sum_{i=1}^{maxskills} G_{n,i} = 1, \forall\, n, where\, n \leq numresources$$

Now, let $TT_{numresources,maxskills}$ be a vector containing the total time spent processing contacts. This vector is updated as follows:

1. Resource r finishes processing a work item wi.
2. Work item wi has skill weight vector BR_{wi}.
3. Total time of processing $T_p$is recorded.

4. TT is updated as

$$TT_{r,n} = TT_{r,n} + T_p, \forall\, n, where\ 0 < n < maxskills\ and\ BR_{wi,n} > 0$$

Now, a total time spent handling each skill is available for each resource. This total time, along with a prospective new work item and percent allocation goals can be used to calculate whether handling this work item helps or hurts the resource in moving toward the allocation goal as follows:

Let $PA_{n,maxskills}$ be the percent allocation for resource n such that:

$$PA_{n,i} = \frac{TT_{n,i}}{\sum_{j=1} maxskillsTT_{n,j}}$$

Now, the current measure of resource n distance from goal DG is calculates as:

$$DG_n = \sum_{i=1}^{maxskills} |PA_{n,i} - G_{n,i}|$$

Now, let T_t be the total time the resource has been handling work items, and let T_{wi} be the total number of work items handled. Also, $T_{avg} = \frac{T_t}{T_{wi}}$. Lastly, let $TTE_{n,maxskills}$ be the estimated total time and $PAE_{n,i}$ be the estimated percent allocation for resource n if it handles work item wi_{new} with skill vector BR_{new} such that:

$$TTE_{n,i} = TT_{n,i} + T_{avg}, \forall\, i, where\ 0 < i < maxskills\ and\ BR_{new,i} > 0$$

$$PAE_{n,i} = \frac{TTE_{n,i}}{\sum_{j=1} maxskillsTTE_{n,j}}$$

Now, our estimated distance from goal DGE for resource n is:

$$DGE_n = \sum_{i=1}^{maxskills} |PAE_{n,i} - G_{n,i}|$$

Now, improvement vs. goal can be expressed as:

$$Improvement = DG_n - DGE_n$$

where negative values indicate moving away from the goal and positive values indicate moving more toward the goal. However, this would cause some ambiguity in later calculations. So, we shift the improvement measurement to insure a positive value as follows:

$$Improvement = DG_n - DGE_n + 2 \times maxskills$$

This yields values between zero and $4 \times maxskills$. We will adjust this value in later calculations for practical use.

4.6 Calculation of Work Item Treatment Value

Assume that the matrix $C_{pendingworkitems,3}$, where *pendingworkitems* is the number of work items awaiting service in the system, represents the work item's current waiting condition by $C_{n,1}$ = number of seconds the work item has been waiting, $C_{n,2}$ = the estimated time that the work item will have to wait for service (see the section of calculating estimated wait for service,) and $C_{n,3}$ = the number of seconds the work item is over its target wait time (or zero if not over acceptable wait time.)

Also, assume that matrix $CW_{pendingworkitems,3}$ represents the weight to be applied to each of the treatment values such that TWn, m is the weight for resource n to be applied to $T_{n,m} \forall m, where 1 \leq m \leq 3$.

Now we calculate the Work Item Treatment value as:

$$WorkItemTreatmentValue_{WorkItem_n} = \sum_{i=1}^{3}(C_{n,i} \times CW_{n,i}) = WTV_n$$

4.7 Calculating Estimated Wait for Service

Calculation of estimated wait for service is accomplished with a moving average of past wait times, weighted across the list of skills and skill requirements as follows:

First, let $F_{maxskills}$ be the vector for holding the moving averages (one moving average for each skill (up to maxskills) and let AW be the average number of work items wating for service and $TWWI$ the Total number of Waiting Work Items . Now, every time a work item wi, having waited time WT, is assigned to a resource, F and AW are updated by:

$$F_i = F_i \times (100 - BR_{wi,i}) + WT \times \frac{BR_{wi,i}}{100}, \forall i, where 1 < i < maxskills$$

$$AW = AW \times 0.9 + TWWI \times 0.1$$

So, not only is this a moving average, it is also a variable moving average with new work items contributing to the moving average between 0 and 10 percent based on their skill needs and weights.

Now, calculating estimated wait for service EWS based on a current work item wi and it's skill requirements vector BRR is done as follows:

$$EWS = \frac{TWWI}{AW} \sum_{i=1}^{maxskills} \frac{BR_{wi,i}}{\sum_{j=1}^{maxskills} BR_{wi,j}} \times F_i$$

Under normal operating conditions, $\frac{TWWI}{AW}$ will be close to 1. However, in the event of a large spike in work volume or a large drop off in work volume, this will adjust the estimate based on slow moving moving averages to more accurately reflect the state of the system.

5 Making the Decision

One final thing remains for resource or work item selection. The wieght of the business value vs. Resource Treatment value, in the case of resource surplus, and the weight of business value vs. Work Item Treatment in case of work item surplus. We will define W_{RSBV} = weight of business value with resource surplus, W_{AT} = weight of Resource Treatment value with resource surplus, W_{WSBV} = weight of business value with work item surplus, and W_{CV} = weight of work item treatment value in the case of work item surplus.

5.1 Resource Surplus and Resource Selection

For the case of resource surplus, we seek to maximize the business value and resource treatment value according to the specified weights by selecting the resource who scores the highest as follows:

$$Max\left[(RSBV_n \times W_{RSBV}) + (RTV_n \times W_{AT})\right], whereA_{n,i} > 0\ \forall BRR_{1,i} > 0.$$

5.2 Work Item Surplus and Work Item Selection

For the case of work item surplus, we seek to maximize the business value and work item treatment value according to the specified weights by selecting the work item which scores the highest as follows:

$$Max\left[(WSBV_n \times W_{WSBV}) + (WTV_n \times W_{CV})\right], whereA_{1,i} > 0\ \forall BRR_{n,i} > 0.$$

6 Scaling Results for Practical Application

The above algorithms select a resource or a work item based on a highest score. However, determining the weights of each of the values that result in an optimum solution for the system can be impractical. For instance, if an resource scores very high in business value ($10 \times 10 = 100$) but the work items have been waiting for an average of 5 minutes, the work item treatment value (WTV) could be as high as 300. So, as the work items age, the WTV has an increased effect on the overall selection. This is undesireable.

So, to be ***practically applicable*** we need to scale the calculation of each of the three selection values. We will need a scaling value for each of the calculations that are performed either for all resources or all work items as follows (denoting scaling values as S):

$$S_{RSBV} = \frac{100}{Max[RSBV_n]}$$

$$S_{WSBV} = \frac{100}{Max[RSBV_n]}$$

$$S_{T_i} = \frac{100}{Max[T_{n,i}]}, 1 \le i \le 3$$

$$S_{C_i} = \frac{100}{Max[C_{n,i}]}, 1 \le i \le 3$$

$$S_T = \frac{100}{Max[T_n]}$$

$$S_C = \frac{100}{Max[C_n]}$$

Now, we can express the weighted values for $RSBV$, $WSBV$, RTV, and WTV as follows:

$$WeightedBusinessValue_{Resource_n} = S_{RSBV} RSBV_n = WRSBV_n$$

$$BusinessValue_{WorkItem_n} = S_{WSBV} WSBVn = WWSBV_n$$

$$PartialWeightedResourceTreatmentValue_{Resource_n} = \sum_{i=1}^{3}(S_{T_i} \times T_{n,i} \times TW_{n,i}) = PWRTV_n$$

$$PartialWeightedWorkItemTreatmentValue_{WorkItem_n} = \sum_{i=1}^{3}(S_{C_i} \times C_{n,i} \times CW_{n,i}) = PWWTV_n$$

$$WeightedResourceTreatmentValue_{Resource_n} = PWRTV_n S_T = WRTV_n$$

$$WeightedWorkItemTreatmentValue_{WorkItem_n} = PWWTV_n S_C = WWTV_n$$

The net result is that we finish the calculation with each of our values ($WRSBV$, $WWSBV$, $WRTV$, and $WWTV$ between 0 and 100. Now, regardless of the units of measure or scaling internal to each of the value's initial configuration we can consistently express the weight that these values have on the overall resource and work item selection.

$$Max\left[(WRSBV_n \times W_{RSBV}) + (WRTV_n \times W_{AT})\right], whereA_{n,i} > 0 \, \forall BRR_{1,i} > 0.$$

$$Max\left[(WWSBV_n \times W_{WSBV}) + (WWTV_n \times W_{CV})\right], whereA_{1,i} > 0 \, \forall BRR_{n,i} > 0.$$

Note that the weighting for the components of Resource Treatment Value ($T_{n,i}$) and Work Item Treatment Value ($C_{n,i}$) are scaled by the above calculations as well.

7 An Example

We will look at an example of resource surplus (Resource Selection) work item-resource matching to clarify the ideas expressed above. Assume that we have two resources, Jane and Jeff. And, we have skills 1-6, English, Chinese, Tech-support, Sales, PC, and UNIX. Jeff speaks English well (skill level 10), speaks Chinese (skill level 5), has been trainied in tech support (skill level 8) and UNIX(skill level 10). Jane speaks English(skill level 10), and is trained and experienced in Sales (skill level 10), she is just learning tech support (skill level

2) and has experience in PC (skill level 7) and UNIX (skill level 6). Also, assume that both Jeff and Jane have logged in and are waiting to receive work items.

Now, our matrix A of available resources looks like this:

$$A = \begin{bmatrix} 10 & 5 & 8 & 0 & 0 & 10 \\ 10 & 0 & 2 & 10 & 7 & 6 \end{bmatrix}$$

Further assume that Jeff has been idle for 1 minute and has been on calls 75% of the day. Jane has been idle for 3 minutes and on calls 50% of the day. Now our matrix T looks like this:

$$T = \begin{bmatrix} 60 & 25 & 0 \\ 180 & 50 & 0 \end{bmatrix}$$

Now, a work item arrives in the system. The work item is qualified with the following requirements: English(weight 10), Tech support (weight 10), and UNIX (weight 5.) All three of these skills are *required*. Now, our matrix BR and matrix BRR look like this:

$$BR = \begin{bmatrix} 10 & 0 & 10 & 0 & 0 & 5 \end{bmatrix}$$

$$BRR = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Lastly, when the work item arrives, it is assigned the following weights:

$$TW = \begin{bmatrix} .25 & .75 & 0 \end{bmatrix}$$

$$W_{RSBV} = .2$$

$$W_{AT} = .8$$

Now, we can calculate our best selection...

$$RSBV_{Jeff} = 230$$

$$RSBV_{Jane} = 150$$

$$WRSBV_{Jeff} = 230\frac{100}{230} = 100$$

$$WRSBV_{Jane} = 150\frac{100}{230} = 65.22$$

$$PWRTV_{Jeff} = \frac{100}{180}(60)(.25) + \frac{100}{50}(25)(.75) + 0 = 45.83$$

$$PWRTV_{Jane} = \frac{100}{180}(180)(.25) + \frac{100}{50}(50)(.75) + 0 = 200$$

$$WRTV_{Jeff} = \frac{100}{200}(45.83) = 22.92$$

$$WRTV_{Jane} = \frac{100}{200} 200 = 100$$

Total Value for Jeff:

$$100(.2) + 22.92(.8) = 40.33$$

Total Value for Jane:

$$65.22(.2) + 100(.8) = 93.044$$

Therefore, Jane is selected to handle the work item. Note that Jeff scored higher in business value and Jane scored higher in agent fairness value. However, due to the 80 % weight on the Resource Fairness value, the work item was delivered to Jane.

8 Interface

In order for another system to use an implementation of the above algorithm, they would need to provide the implementation with the following:

8.1 Adminstration

Some facilities must be created for adminstering the agents and their skill levels for each of the skills.

8.2 Resource State Change Events

The implementation of the algorithm would need to provide an interface through which other applications could update the state of resources as necessary. For example, as agents become available in a system, the application monitoring the agents would send a *Resource Ready* message to the algorithm for notification that the resource is available to handle work items.

8.3 Arriving Work Item Events

Each arriving work item must be handed to the algorithm with the following information

- The vector of skill weights (BR)
- The vector of skill requirement (BRR)
- The weight of business value with resource surplus (W_{RSBV})
- The weight of business value with work item surplus (W_{WSBV})
- The vector of resource treatment weights (W_{AT})
- The vector of work item treatment weights (W_{CV})
- The target wait time for this work item ($Target$)

Attachment D

BCS-DENVER PATENT COMMITTEE MEETING - February 11, 1999   SUBMISSIONS DR-PAT-99-01 thru DR-PAT-99-19

A New Algorithm for Resource and Work Item Selection

Jon Ford, Lucent Technologies

April 15, 1999

1 Overview

Moving resource and contact selection algorithms off the switch has opened the door to many new opportunities for higher optimization and flexibility. The algorithm described below was initially developed for distributing calls in a call center. However, as the ideas were developed it became more and more obvious that these ideas apply to any kind of work distribution. So, the algorithm described below is a more general solution to what started out as a specific problem.

2 Current Limitations

2.1 Hardware Limitations

Traditional call center resource and work item selection algorithms have been limited by the *call center model* available in the switch. For example, Definity uses resources with skills and skill levels. The model then provides multi-priority queues for queueing work items to wait for the resources with those skills to become available. So, any algorithms we apply to this model are limited by its unflexible and simple structure. The limitations become more obvious when we look at an example in the following section.

2.2 Example - Acme Insurance

Supose that Acme Insurance Company sells 3 types of insurance: auto, home, and life. Also, suppose that each resource must be licenced to sell each of these insurance policies on a state by state basis. Lastly, suppose that the call center allows callers to select English or Spanish speaking resources. This results in $3 \times 50 \times 2 = 300$ possible combinations of caller skill requirements. Administering resources and skills using these combinations would be difficult at best. In fact, given the current limitations with market leading ACDs, a work-around must be found.

3 A New Approach

Separated from the constraints of the legacy switching systems, we are free tc apply *common sense* to our way of thinking about resources, skills, and callers. We will define resources with skills and proficiencies in those skills. We will qualify calls (work items) when they arrive in the system as requiring sets of skills. We will set goals for work item handling and resource treatment. Lastly, we will optimize and allow extensive customization of the process of matching the resources with the work items - even on a work item or resource specific basis.

4 The Algorithm

The goal of the algorithm is to match work items and resources in such a way that it brings the most value to all of the stakeholders in the call center organization. This includes customers, resources, managers, etc. To perform this optimization, we will evaluate three components of the work item-resource match.

4.1 The Three System Values

4.1.1 Business Value

The *Business Value* of a work item-resource match is a measure of resource qualification for work item handling based on resource skills and skill proficiencies and work item skill requirements.

4.1.2 Resource Treatment Value

The *Resource Treatment Value* of a work item-resource match is a measure of how an resource is spending time compared with other resources as well as individual resource goals.

4.1.3 Work Item Treatment Value

The *Work Item Treatment Value* of a work item-resource match is a measure of how a work item is treated compared to other work items as well as individual treatement goals for the specific work item.

4.2 The Two Basic System States

Evaluation of the Three System Values discussed above depends on the state of the system at the time of evalution. The two possible states for a system and their effect on the application of the Three System Values is described in the following sections.

4.2.1 Resource Surplus

If there are resources working in the system who are able to handle work items but are waiting for a work item, the system is considered to be in a state of *Resource Surplus*. In the case of resource surplus, when a work item arrives in the system, the objective is to find the *best* resource to handle the work item. So, we would evaluate the Business Value and the Resource Treatment Value of each resource matched with the new work item, looking for the resource-work item match that brought the highest value to the organization. For this case of Resource Surplus, Work Item Treatment is not an issue as the work item will receive immediate handling, provided that there is an resource available with the required skills.

4.2.2 Work Item Surplus

If there are no resources in the system who are able to handle work items and there are work items waiting for resource assignment, the system is considered to be in a state of *Work Item Surplus*. In the case of work item surplus, when a resource becomes available to handle work items, the objective is to find the *best* work item for the resource to handle. So, we would evaluate the Business Value and the Work Item Treatment value of each work item matched with the newly available resource, looking for the resource-work item match that brings the hightest value to the organization. For the case of Resource Surplus, Resource Treatment is not an issue as the resource will receive the work item immediately upon becoming available, provided that there is a work item available for which the resource has the required skills.

4.3 Calculation of Business Value

Assume that matrix $A_{availresources,maxskills}$, where $availresources$ is the number of available resources and $maxskills$ is the maximum number of skills defined in the system, represents the available resources and their associated skill levels such that $A_{n,m}$ represents the skill level (an integer between 0 and 10) for resource n and skill m.

Also, assume that matrix $BR_{pendingworkitems,maxskills}$, where $pendingworkitems$ is the number of unserviced work items in the system and $maxskills$ is the maximum number of skills defined in the system, represents the skill *weight* of the work item such that $BR_{n,m}$ represents the skill *weight* (an integer between 0 and 10) for work item n and skill m.

Lastly, assume that matrix $BRR_{pendingworkitems,maxskills}$, where $pendingworkitems$ is the number of unserviced work items in the system and $maxskills$ is the maximum number of skills defined in the system, represents the *requirement* of a skill for a work item such that $BRR_{n,m}$ (true of false) indicates whether a resource *must* have a skill level > 0 in skill m to handle work item n. This array is not used in the calculation of business value, but it will be used in following sections.

4.3.1 Resource Surplus

In the case of Resource Surplus System State, the business value is calculated as:

$$BusinessValue_{Resource_n} = \sum_{i=1}^{maxskills} (A_{n,i} \times BR_{1,i}) = RSBV_n$$

4.3.2 Work Item Surplus

In the case of Work Item Surplus System State, the business value is calculates as:

$$BusinessValue_{WorkItem_n} = \sum_{i=1}^{maxskills} (A_{1,n} \times BR_{n,i}) = WSBV_n$$

4.4 Calculation of Resource Treatment Value

Assume that the matrix $T_{availresources,3}$, where $availresources$ is the number of resources in the system waiting to service work items, represents the resource's activity during the current login session by $T_{n,1}$ = time in seconds since becoming available, $T_{n,2}$ = 100 - percent of logged in time spent handling work items and their work items associated after call work, and $T_{n,3}$ = a measure of how much handling the current work item would move them toward their goal (See section on calculation and setting of resource goals.)

Next, assume that matrix $TW_{pendingworkitems,3}$ represents the weight to be applied to each of the treatment values such that $TW_{n,m}$ is the weight for resource n to be applied to $T_{n,m} \forall m, where 1 \leq m \leq 3$.

Now, we calculate the Resource Treatment value as:

$$ResourceTreatmentValue_{Resource_n} = \sum_{i=1}^{3} (T_{n,i} \times TW_{n,i}) = RTV_n$$

4.5 Calculating and Setting Resource Goals

Let $G_{maxskills,numresources}$ be a vector containing percent allocation goals for all resources up to maxskills such that

$$\sum_{i=1}^{maxskills} G_{n,i} = 1, \forall n, where\ n \leq numresources$$

Now, let $TT_{numresources,maxskills}$ be a vector containing the total time spent processing contacts. This vector is updated as follows:

1. Resource r finishes processing a work item wi.
2. Work item wi has skill weight vector BR_{wi}.
3. Total time of processing T_p is recorded.

4. TT is updated as

$$TT_{r,n} = TT_{r,n} + T_p, \forall\, n, where\ 0 < n < maxskills\ and\ BR_{wi,n} > 0$$

Now, a total time spent handling each skill is available for each resource. This total time, along with a prospective new work item and percent allocation goals can be used to calculate whether handling this work item helps or hurts the resource in moving toward the allocation goal as follows:

Let $PA_{n,maxskills}$ be the percent allocation for resource n such that:

$$PA_{n,i} = \frac{TT_{n,i}}{\sum_{j=1} maxskillsTT_{n,j}}$$

Now, the current measure of resource n distance from goal DG is calculates as:

$$DG_n = \sum_{i=1}^{maxskills} |PA_{n,i} - G_{n,i}|$$

Now, let T_t be the total time the resource has been handling work items, and let T_{wi} be the total number of work items handled. Also, $T_{avg} = \frac{T_t}{T_{wi}}$. Lastly, let $TTE_{n,maxskills}$ be the estimated total time and $PAE_{n,i}$ be the estimated percent allocation for resource n if it handles work item wi_{new} with skill vector BR_{new} such that:

$$TTE_{n,i} = TT_{n,i} + T_{avg}, \forall\, i, where\ 0 < i < maxskills\ and\ BR_{new,i} > 0$$

$$PAE_{n,i} = \frac{TTE_{n,i}}{\sum_{j=1}^{maxskills} TTE_{n,j}}$$

Now, our estimated distance from goal DGE for resource n is:

$$DGE_n = \sum_{i=1}^{maxskills} |PAE_{n,i} - G_{n,i}|$$

Now, improvement vs. goal can be expressed as:

$$Improvement = DG_n - DGE_n$$

where negative values indicate moving away from the goal and positive values indicate moving more toward the goal. However, this would cause some ambiguity in later calculations. So, we shift the improvement measurement to insure a positive value as follows:

$$Improvement = DG_n - DGE_n + 2 \times maxskills$$

This yields values between zero and $4 \times maxskills$. We will adjust this value in later calculations for practical use.

4.6 Calculation of Work Item Treatment Value

Assume that the matrix $C_{pendingworkitems,3}$, where *pendingworkitems* is the number of work items awaiting service in the system, represents the work item's current waiting condition by $C_{n,1}$ = number of seconds the work item has been waiting, $C_{n,2}$ = the estimated time that the work item will have to wait for service (see the section of calculating estimated wait for service,) and $C_{n,3}$ = the number of seconds the work item is over its target wait time (or zero if not over acceptable wait time.)

Also, assume that matrix $CW_{pendingworkitems,3}$ represents the weight to be applied to each of the treatment values such that TWn, m is the weight for resource n to be applied to $T_{n,m} \forall m, where 1 \leq m \leq 3$.

Now we calculate the Work Item Treatment value as:

$$WorkItemTreatmentValue_{WorkItem_n} = \sum_{i=1}^{3} (C_{n,i} \times CW_{n,i}) = WTV_n$$

4.7 Calculating Estimated Wait for Service

Calculation of estimated wait for service is accomplished with a moving average of past wait times, weighted across the list of skills and skill requirements as follows:

First, let $F_{maxskills}$ be the vector for holding the moving averages (one moving average for each skill (up to maxskills) and let AW be the average number of work items wating for service and $TWWI$ the Total number of Waiting Work Items . Now, every time a work item wi, having waited time WT, is assigned to a resource, F and AW are updated by:

$$F_i = F_i \times (100 - BR_{wi,i}) + WT \times \frac{BR_{wi,i}}{100}, \forall i, where 1 < i < maxskills$$

$$AW = AW \times 0.9 + TWWI \times 0.1$$

So, not only is this a moving average, it is also a variable moving average with new work items contributing to the moving average between 0 and 10 percent based on their skill needs and weights.

Now, calculating estimated wait for service EWS based on a current work item wi and it's skill requirements vector BRR is done as follows:

$$EWS = \frac{TWWI}{AW} \sum_{i=1}^{maxskills} \frac{BR_{wi,i}}{\sum_{j=1}^{maxskills} BR_{wi,j}} \times F_i$$

Under normal operating conditions, $\frac{TWWI}{AW}$ will be close to 1. However, in the event of a large spike in work volume or a large drop off in work volume, this will adjust the estimate based on slow moving moving averages to more accurately reflect the state of the system.

5 Making the Decision

One final thing remains for resource or work item selection. The wieght of the business value vs. Resource Treatment value, in the case of resource surplus, and the weight of business value vs. Work Item Treatment in case of work item surplus. We will define W_{RSBV} = weight of business value with resource surplus, W_{AT} = weight of Resource Treatment value with resource surplus, W_{WSBV} = weight of business value with work item surplus, and W_{CV} = weight of work item treatment value in the case of work item surplus.

5.1 Resource Surplus and Resource Selection

For the case of resource surplus, we seek to maximize the business value and resource treatment value according to the specified weights by selecting the resource who scores the highest as follows:

$$Max\,[(RSBV_n \times W_{RSBV}) + (RTV_n \times W_{AT})]\,, where A_{n,i} > 0\ \forall BRR_{1,i} > 0.$$

5.2 Work Item Surplus and Work Item Selection

For the case of work item surplus, we seek to maximize the business value and work item treatment value according to the specified weights by selecting the work item which scores the highest as follows:

$$Max\,[(WSBV_n \times W_{WSBV}) + (WTV_n \times W_{CV})]\,, where A_{1,i} > 0\ \forall BRR_{n,i} > 0.$$

5.3 Optional: Dynamic Business Value and Work Item Treatment Weights

As an option to setting predefined weights for business value and work item treatment value in the case of work item surplus, a more effective solution would allow the values to change over time. This change would increase the weight of the work item treatment value over time in order to prevent starvation and move closer to the work item's target service time. In this scenario, work items would have initial values of $W_{RSBV} = 1$ and $W_{CV} = 0$. These values would then change over time (during each evalutaion) as:

$$W_{CV} = \frac{currentwaittime}{targetwaittime}$$

or 1, whichever is greater

$$W_{RSBV} = 1 - W_{CV}$$

This would start the calculation with the most weight on the skills of the resource and move to applying the most weight to insuring fair treatment of the work item.

6 Scaling Results for Practical Application

The above algorithms select a resource or a work item based on a highest score. However, determining the weights of each of the values that result in an optimum solution for the system can be impractical. For instance, if an resource scores very high in business value ($10 \times 10 = 100$) but the work items have been waiting for an average of 5 minutes, the work item treatment value (WTV) could be as high as 300. So, as the work items age, the WTV has an increased effect on the overall selection. This is undesireable.

So, to be *practically applicable* we need to scale the calculation of each of the three selection values. We will need a scaling value for each of the calculations that are performed either for all resources or all work items as follows (denoting scaling values as S):

$$S_{RSBV} = \frac{100}{Max[RSBV_n]}$$

$$S_{WSBV} = \frac{100}{Max[RSBV_n]}$$

$$S_{T_i} = \frac{100}{Max[T_{n,i}]}, 1 \leq i \leq 3$$

$$S_{C_i} = \frac{100}{Max[C_{n,i}]}, 1 \leq i \leq 3$$

$$S_T = \frac{100}{Max[T_n]}$$

$$S_C = \frac{100}{Max[C_n]}$$

Now, we can express the weighted values for $RSBV$, $WSBV$, RTV, and WTV as follows:

$$WeightedBusinessValue_{Resource_n} = S_{RSBV} RSBV_n = WRSBV_n$$

$$BusinessValue_{WorkItem_n} = S_{WSBV} WSBVn = WWSBV_n$$

$$PartialWeightedResourceTreatmentValue_{Resource_n} = \sum_{i=1}^{3} (S_{T_i} \times T_{n,i} \times TW_{n,i}) = PWRTV_n$$

$$PartialWeightedWorkItemTreatmentValue_{WorkItem_n} = \sum_{i=1}^{3} (S_{C_i} \times C_{n,i} \times CW_{n,i}) = PWWTV_n$$

$$WeightedResourceTreatmentValue_{Resource_n} = PWRTV_n S_T = WRTV_n$$

$$WeightedWorkItemTreatmentValue_{WorkItem_n} = PWWTV_n S_C = WWTV_n$$

The net result is that we finish the calculation with each of our values ($WRSBV$, $WWSBV$, $WRTV$, and $WWTV$ between 0 and 100. Now, regardless of the units of measure or scaling internal to each of the value's initial

configuration we can consistently express the weight that these values have on the overall resource and work item selection.

$$Max\left[(WRSBV_n \times W_{RSBV}) + (WRTV_n \times W_{AT})\right], where A_{n,i} > 0 \, \forall BRR_{1,i} > 0.$$

$$Max\left[(WWSBV_n \times W_{WSBV}) + (WWTV_n \times W_{CV})\right], where A_{1,i} > 0 \, \forall BRR_{n,i} > 0.$$

Note that the weighting for the components of Resource Treatment Value ($T_{n,i}$) and Work Item Treatment Value ($C_{n,i}$) are scaled by the above calculations as well.

7 An Example

We will look at an example of resource surplus (Resource Selection) work item-resource matching to clarify the ideas expressed above. Assume that we have two resources, Jane and Jeff. And, we have skills 1-6, English, Chinese, Tech-support, Sales, PC, and UNIX. Jeff speaks English well (skill level 10), speaks Chinese (skill level 5), has been trainied in tech support (skill level 8) and UNIX(skill level 10). Jane speaks English(skill level 10), and is trained and experienced in Sales (skill level 10), she is just learning tech support (skill level 2) and has experience in PC (skill level 7) and UNIX (skill level 6). Also, assume that both Jeff and Jane have logged in and are waiting to receive work items.

Now, our matrix A of available resources looks like this:

$$A = \begin{bmatrix} 10 & 5 & 8 & 0 & 0 & 10 \\ 10 & 0 & 2 & 10 & 7 & 6 \end{bmatrix}$$

Further assume that Jeff has been idle for 1 minute and has been on calls 75% of the day. Jane has been idle for 3 minutes and on calls 50% of the day. Now our matrix T looks like this:

$$T = \begin{bmatrix} 60 & 25 & 0 \\ 180 & 50 & 0 \end{bmatrix}$$

Now, a work item arrives in the system. The work item is qualified with the following requirements: English(weight 10), Tech support (weight 10), and UNIX (weight 5.) All three of these skills are *required*. Now, our matrix BR and matrix BRR look like this:

$$BR = \begin{bmatrix} 10 & 0 & 10 & 0 & 0 & 5 \end{bmatrix}$$

$$BRR = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Lastly, when the work item arrives, it is assigned the following weights:

$$TW = \begin{bmatrix} .25 & .75 & 0 \end{bmatrix}$$

$$W_{RSBV} = .2$$

$$W_{AT} = .8$$

Now, we can calculate our best selection...

$$RSBV_{Jeff} = 230$$

$$RSBV_{Jane} = 150$$

$$WRSBV_{Jeff} = 230\frac{100}{230} = 100$$

$$WRSBV_{Jane} = 150\frac{100}{230} = 65.22$$

$$PWRTV_{Jeff} = \frac{100}{180}(60)(.25) + \frac{100}{50}(25)(.75) + 0 = 45.83$$

$$PWRTV_{Jane} = \frac{100}{180}(180)(.25) + \frac{100}{50}(50)(.75) + 0 = 200$$

$$WRTV_{Jeff} = \frac{100}{200}(45.83) = 22.92$$

$$WRTV_{Jane} = \frac{100}{200}200 = 100$$

Total Value for Jeff:

$$100(.2) + 22.92(.8) = 40.33$$

Total Value for Jane:

$$65.22(.2) + 100(.8) = 93.044$$

Therefore, Jane is selected to handle the work item. Note that Jeff scored higher in business value and Jane scored higher in agent fairness value. However, due to the 80 % weight on the Resource Fairness value, the work item was delivered to Jane.

8 Resource Skill Adminstration

The above example demonstrates some of the uses of the algorithm. However, the algorithm is capable of much more. In this section we will look more specifically at two types of resource skills and how to administer those skills so the algorithm performs to more of its full potential.

8.1 Unitary Skills

Unitary skills are skills with which we are most familiar. Skills such as English, Spanish, Sales, Tech. Support, UNIX, Windows, and Macintosh fall into this category. These are also the most simple to administer. Each skill is assigned a number starting at 1 such that $A_{n,m}$ represents the skill level of resource n for the skill assigned to position m - just as in our example above.

8.2 Aggregate Skills

Aggregate skills are slightly more difficult to work with but are simple nonetheless. They are skills that are made up of combinations of unitary skills as described above. Aggregate skills are needed in that there may be a resource capable of handling several skills individually *and* only in certain combinations. For instance, suppose there is an agent in a call center who speaks Spanish, knows UNIX, Windows, and Macintosh. Also, suppose that this agent has had some technical support training and therefore is skilled in tech support as well. However, this agent is being *eased into* the tech support role. For the first two weeks the agent should only handle Spanish tech support calls for Windows. During the following two weeks, the agent can also take Spanish tech support calls for Macintosh. Finally, the agent is also allowed to handle Spanish tech support calls for UNIX. This scenario is most easily adminstered with three *aggregate skills* : Spanish Windows Tech Support, Spanish Macintosh Tech Support, and Spanish UNIX Tech Support.

Now, more generally, aggregate skills are adminstered as follows. Suppose that some aggregate skill g is an aggregate of unitary skills a, b, and c. Now, $A_{n,g}$ represents the skill level of resource n for aggregate skill g, where $g > maxunitskills$, and $maxunitskills$ is the total number of unitary skills.

This allows for further differentiation of resources by ability to handle not only individual skill requirements but skill requirement combinations as well.

9 Work Item Qualification

In order to evaluate a work item and distribute it to the *best* resource, it must be presented to the system fully qualified according to its specific needs for service. The details of what needs to be provided are outlined below.

9.1 Qualifying for Calcuation of Business Value

Each skill required or simply desired for the handling of of a work item must be *added* to the work item qualification for each of the categories below.

9.1.1 Unitary Skills

For each unitary skill associated with work item n, a weight (integer between 0 and 10) should be assigned such that $BR_{n,m} = w$ where $1 \leq m \leq maxunitskills, 0 \leq w \leq 10$, and $maxunitskills$ is the number of unitary skills.

9.1.2 Aggregate Skills

For each aggregate skill associated with work item n, a weight (integer between 0 and 10) should be assigned such that $BR_{n,m} = w$ where $maxskills < m \leq maxskills, 0 \leq w \leq 10$, and $maxskills$ is the total number of unitary and aggregate skills combined.

9.1.3 Skill Requirements

For each skill, both aggregate and unitary, assocated with work item n, a value of true or false must be assigned to BRR such that $BRR_{n,m}$ indicates whether or not a skill m is *required* for work item n. Note that it is possible for a skill to contribute to the business value calculation but not be a *required* skill. For example, suppose that some agents in a call center had been trained in effective communication but this is not *required* to handle a call. Those agents could score higher in business value, having had that training. However, agents who were not trained would not be excluded from the selection process if the skill had a weight > 0 in BRR but not a value of true in the requirements vector BRR.

9.2 Qualifying Weights for Resource Fairness Values

The values, for resource n, of $TW_{n,m}$ must be filled in for fainess values one through three ($1 \leq m \leq 3$.) Remember from the section on Calculation of Resource Treatment Value that the three values are time since becoming available, 100 - percent time spent handling work items, and how much handling the current work item moves the resource toward its goal. Although the values of TW can be anything, it is most appropriate to imagine that $\sum_{i=1}^{i=3} TW_{n,i} = 1$ for a *percentage weight* of each of the values for some resource n.

9.3 Qualifying Weights for Work Item Fairness Values

As with resources above, the values, for work item n, of $CW_{n,m}$ must be filled in for fainess values one through three ($1 \leq m \leq 3$.) Remember from the section on Calculation of Work Item Treatment Value that the three values are time waiting, estimated total wait time, and time over targer wait time. Although the values of CW can be anything, it is most appropriate to imagine that $\sum_{i=1}^{i=3} TW_{n,i} = 1$ for a *percentage weight* of each of the values for some resource n.

9.4 Qualifying Weights for Business Value

When a new work item is passed into the system for evaluation and distribution, it is assumed that the source of the work item knows nothing of the state of the system and must provide enough information to make a decision in any state.

9.4.1 Resource Surplus

So, to be able to make a decision in a resource surplus state, the weight of business value with resource surplus, W_{RSBV}, must be provided.

9.4.2 Work Item Surplus

And, to be able to make a decision in a work item surplus state, the weight of business value with work item surplus, W_{WSBV}, must be provided.

9.5 Qualifying Target Wait Time

Lastly, the target wait time for the work item, $Target$, must also be provided.

10 Additional Required Qualification

In order for another system to use an implementation of the above algorithm, they would need to provide the implementation with the following:

10.1 Adminstration

Some facilities must be created for adminstering the agents and their skill levels for each of the skills as described in the sections above.

10.2 Resource State Change Events

The implementation of the algorithm would need to provide an interface through which other applications could update the state of resources as necessary. For example, as agents become available in a system, the application monitoring the agents would send a *Resource Ready* message to the algorithm for notification that the resource is available to handle work items.

10.3 Arriving Work Item Events

Each arriving work item must be handed to the algorithm with the information outlined in the section on Work Item Qualification above.

10.4 Interfaces for Delivery of Work Items to Resources

The program implementing the algorithm described above would need to have available an interface for delivering work items to agents. For example, if the program were controlling the distribution of calls in a call center, there would need to be a CTI interface available to the program to allow for the tranfer of calls to agents.

11 Notes About a Distributed Solution

The compact nature of this algorithm allows for distribution across multiple sites. It would be quite effective for multiple sites to update the current values of their forecast vector F, their average number of waiting work items, AW, the number of current waiting work items $TWWI$, and a vector indicating

servicable skills (to avoid sending a work item to a site that couldn't service it) at a central location. This central location would then have enough information to distribute work items across the multiple sites based on estimated time to service. This problem needs further exploration.

12 Wish List

- Predict Wait time by work item history if available for the work item source. For example, known history for a caller to a call center.
- Simulation to determine effective weights and weight ranges.
- Simulation to show algorithm effectiveness.
- Simulation to show *distributability* of the algorithm.

Lucent Technologies Inc.
Intellectual Property-Law

Attachment F
Page 1

INVENTION RECORD

From: The Undersigned

To: David Volejnicek

Case Name/Number: J.A. Ford 1

Title: Workflow Resource And Work Item Selection Arrangement

Identify document(s) which describe the invention.
Document(s) and Date(s):

A New Algorithm for Resource and Work Item Selection
April 15, 1999

Describe the early history of the invention, including dates and documents not identified above.
History, Document(s) and Date(s):

The idea was developed from January 1999 – March 1999 and a prototype software implementation was created in March 1999.

Identify any person other than a co-inventor who was initially made aware of the invention.
Name(s) and Date(s):

Dave Bauer	Feb. 1999
Ramesh Achuthan	Feb. 1999
Jeff Chin	Feb. 1999

Identify any documents describing the invention that were read or witnessed by any such person(s).
Document(s), Name(s), and Date(s):

Document Describing invention above.

Attach copies, where possible, of each of the documents described above.

Describe what was implemented or demonstrated in connection with the invention.

A prototype software implementation was created in March 1999.

Identify person(s) who worked on the implementation/demonstration.
Name(s) and Date(s):

Jon Ford Feb 1999 - March 1999

Identify person(s) who tested the implementation or carried out the demonstration.
Name(s) and Date(s):

Jon Ford March 1999

Identify person(s) who witnessed or observed the implementation and/or tests.
Name(s) and Date(s):

Ramesh Achuthan March 1999

Identify any documents describing the implementation and/or tests.

Software only (Java Code) viewable at
http://alfred.dr.lucent.com/cgi-bin/cvsweb/bw/prototypes/distrib/

Inventor(s) signature(s) and Date(s):

10/21/99

Attach copies, where possible of each of the documents described above.

Attachment G

Lucent Technologies
Bell Labs Innovations

subject: **Patent IDS No. 119349**
Workflow Resource And Work Item Selection Arrangement (DR-PAT-99-18)

date: **June 24, 1999**

from: **BCS-Denver Patent Committee**

J. P. Anderson:

On February 1, 1999, the BCS-Denver Patent Committee received a submission from J. A. Ford characterizing a "Workflow Resource And Work Item Selection Arrangement". This submission was forwarded to the Intellectual Property division to study the patentability aspects of this proposal.

We have now concluded our patentability study of this submission and have determined that it appears to contain sufficient novelty to warrant seeking patent protection thereon. We have also concluded that the inventor is J. A. Ford. Accordingly, this application will be designated as J. A. Ford 1. This determination has been discussed with the inventor, who concurs.

David Volejnicek

David Volejnicek
Corporate Counsel
(for the BCS-Denver Patent Committee)

DR-P33A20000-DV-lw

Copy to
J. Y. Payseur

J. A. Ford 1

PATENT APPLICATION DATA

Title: Workflow Resource And Work Item Selection Arrangement

Attorney: David Volejnicek (303-538-4154)

Inventors: Jon Allan Ford LUCENT cof110/80881 139KE0000

First Filed Country: United States of America

Type Appln: Utility
Kind Appln: New
Related Appln(s):
Assignee(s): LUCENT
Portfolio(s):
Classification Code History:

Code	Date Assigned	Decision Made By
II	06/24/1999	DR Patent Committee

BU Code(s): REAL

IDS Number(s): 119349
IDA Number: 290984
Use of Invention:

Mailed:
Filed:
Serial No:

Export Lic Attn: No
Gov't Contract: No

Abstract:

Foreign Filing Decision: YES

Countries	N	P/N	P/E	E
Canada	X			
Japan	X			
Republic of Korea	X			
Great Britain				X
France				X
Fed. Rep. of Germany				X

Printed: 6/24/99